APPROXIMATE REGULARIZATION PATHS FOR NUCLEAR NORM MINIMIZATION USING SINGULAR VALUE BOUNDS – WITH IMPLEMENTATION AND EXTENDED APPENDIX

N. Blomberg[†], C.R. Rojas[†], B. Wahlberg[†]

[†]Department of Automatic Control and ACCESS Linnaeus Center, School of Electrical Engineering, KTH–Royal Institute of Technology, SE-100 44 Stockholm, Sweden. {nibl, crro, bo}@kth.se.

ABSTRACT

The widely used nuclear norm heuristic for rank minimization problems introduces a regularization parameter which is difficult to tune. We have recently proposed a method to approximate the regularization path, i.e., the optimal solution as a function of the parameter, which requires solving the problem only for a sparse set of points. In this paper, we extend the algorithm to provide error bounds for the singular values of the approximation. We exemplify the algorithms on large scale benchmark examples in model order reduction. Here, the order of a dynamical system is reduced by means of constrained minimization of the nuclear norm of a Hankel matrix.

Index Terms— Nuclear norm heuristic, regularization path, singular value perturbation, model order reduction.

1. INTRODUCTION

Rank minimization has important applications in e.g. signal processing, control, machine learning, system identification, and model order reduction. The matrix argument can e.g. be a covariance matrix (as in sensor array processing and multivariate statistical data analysis) or a structured matrix such as a Hankel matrix (as in system realization), [?]. Specifically, application areas include spectrum sensing [?], signal time delay estimation [?], phase retrieval of sparse signals [?], wireless network inference [?], channel equalization [?], etc.

In general, the rank minimization problem is non-convex and NP-hard [?]. However, a common convex heuristic for these problems is nuclear norm minimization. The nuclear norm $\|\cdot\|_* = \sum_i \sigma_i(\cdot)$, i.e., the sum of the singular values, is used as a convex surrogate for the non-convex rank function; this is so because the nuclear norm can be interpreted as a convex relaxation of the rank, since it is the pointwise tightest convex function (called a *convex envelope* [?]) to lower-bound the rank, for matrices inside a unit spectral-norm ball.

Consider a general case of minimization of the nuclear norm of a linear map subject to a quadratic constraint:

$$
\begin{aligned}
\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad & \|\mathcal{A}(x)\|_* \\
\text{subject to} \quad & \|x - x_o\|_2 \leq \lambda,
\end{aligned}
\tag{1}
$$

where $\mathcal{A} : \mathbb{R}^n \to \mathbb{R}^{p \times q}$ is a linear map (for simplicity, from now on we treat the symmetric case, $p = q$), $x \in \mathbb{R}^n$ is the decision variable, and λ is the regularization parameter.

Note that the formulation in (1) belongs to a subclass of regularized nuclear norm optimization problems. Other formulations include exchanging cost and constraint or the penalized version [?]. In addition, our theory can readily be extended to weighted norms, $\|x\|_W := x^T W x$. Then, the quadratic constraint is equivalent to the general quadratic inequality $x^T P x + q^T x + r \leq 0$.

The key issue here is that, although regularized nuclear norm minimization has been thoroughly studied, it suffers from the fact that the dependence of the solution on the regularization parameter is difficult to predict. Without, in general, *a priori* knowledge on how to choose λ, we are motivated to study the so called *regularization path*, i.e., the optimal solution as a function of the regularization parameter. For problem (1) the regularization path is defined on the domain

$$
\lambda \in (\lambda_{\min}, \lambda_{\max}) := (0, \|x_o\|_2),
\tag{2}
$$

since for $\lambda = 0$ the solution to (1) is known, $x^{\text{opt}} = x_o$, and for $\lambda \geq \|x_o\|_2$ the constraint set is large enough to include the unconstrained minimum, $x^{\text{opt}} = 0$.

For practical purposes the domain of the regularization path must be discretized, which raises the question of how to choose the grid points. This is indeed an important question since problem (1) can be computationally costly to solve.

To address this problem, in [?], we presented a method to choose the grid points based on a worst-case approximation error when the optimal solution for λ, x_λ^{opt}, is approximated by $x_{\lambda^\star}^{\text{opt}}$ for $\lambda^\star < \lambda$. The idea is visualized in Figure 1. Given the solution for some $\lambda^\star$, we increase λ beyond $\lambda^\star$ until the worst-case approximation error reaches a prespecified tolerance, ε, and then we re-evaluate (1). Iteratively, starting for $\lambda_0^\star = 0$, this procedure generates a set of grid points, $\lambda_i^\star, i = 1, \ldots, m$, and an approximate regularization path such that the approximation error is within ε for all λ.
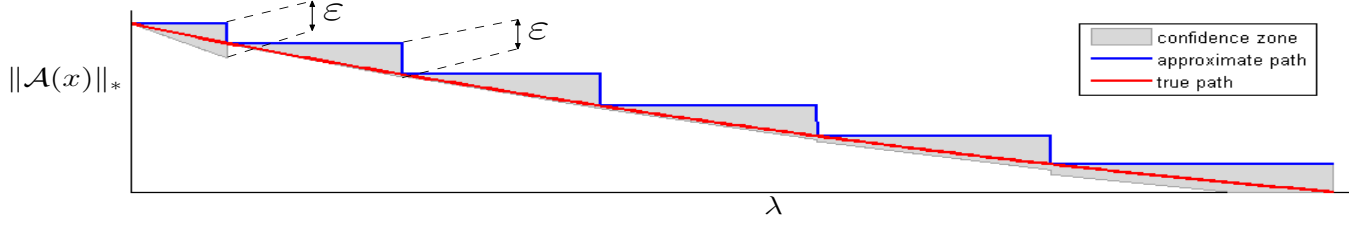
Fig. 1. Illustration of regularization path algorithm proposed in [**?**]. x-axis: regularization parameter, λ. y-axis: cost of (1). The true regularization path (red) is guaranteed to lie in the shaded zone. The approximate path (blue) is guaranteed to differ by at most ε from the true path.

The novelty of this paper consists of two new algorithms. The first gives a guarantee on the cost function of (1). The second gives a guarantee on the singular values of $\mathcal{A}(x_\lambda^{\mathrm{opt}})$, when x_λ^{opt} is approximated by $x_{\lambda^\star}^{\mathrm{opt}}$. Furthermore, we derive upper bounds on the number of grid points needed by the algorithms to meet a tolerance ε.

2. ERROR BOUNDS FOR APPROXIMATION OF (1)

In this section we derive error bounds that allow us to confine the true regularization path within a certain region (the shaded area in Figure 1).

Define the singular values of $\mathcal{A}(x_\lambda^{\mathrm{opt}})$, where x_λ^{opt} is optimal for (1) for parameter value λ, as

$$\sigma\left(\mathcal{A}\left(x_\lambda^{\mathrm{opt}}\right)\right) =: \left(\sigma_1^\lambda, \ldots, \sigma_p^\lambda\right).$$

For further use in the below presented Algorithms 1 and 2, respectively, we derive upper bounds on the quantities:

$$\left\|\mathcal{A}\left(x_{\lambda^\star}^{\mathrm{opt}}\right)\right\|_* - \left\|\mathcal{A}\left(x_\lambda^{\mathrm{opt}}\right)\right\|_* \text{ and } \sum_i^p \left(\sigma_i^{\lambda^\star} - \sigma_i^\lambda\right)^2, \quad (3,4)$$

where $x_{\lambda^\star}^{\mathrm{opt}}$ is given. The bounds can be viewed as worst-case approximation errors in the singular values when x_λ^{opt} is approximated by $x_{\lambda^\star}^{\mathrm{opt}}$.

2.1. Relaxation of (1) using subgradients

We here relax problem (1) using subgradients of the nuclear norm. The concept of subdifferentials (or sets of subgradients) is a generalization of the gradient that applies to functions whose gradient is undefined in some point or points, [**?**]. In the case of the nuclear norm, the subdifferential is (see e.g. [**?**]):

$$\partial \|X\|_* = \left\{UV^T + W : U^TW = WV = 0, \|W\| \leq 1\right\},$$

where $X = U\Sigma V^T \in \mathbb{R}^{p \times p}$ is a compact singular value decomposition $W \in \mathbb{R}^{p \times p}$. $U^TW = W^TV = 0$ implies that X and W must have orthogonal row and column spaces.

Now, assume that $x_{\lambda^\star}^{\mathrm{opt}}$ solves (1) for some parameter value $\lambda = \lambda^\star$. Then, since the nuclear norm is convex we can write, for any matrix $\mathcal{A}(x) \in \mathbb{R}^{p \times p}$, the inequality

$$\|\mathcal{A}(x)\|_* \geq \left\|\mathcal{A}\left(x_{\lambda^\star}^{\mathrm{opt}}\right)\right\|_* + \left\langle U_{\lambda^\star}V_{\lambda^\star}^T + W, \mathcal{A}(x) - \mathcal{A}\left(x_{\lambda^\star}^{\mathrm{opt}}\right)\right\rangle$$
$$= \left\|\mathcal{A}\left(x_{\lambda^\star}^{\mathrm{opt}}\right)\right\|_* + \mathcal{A}^*(U_{\lambda^\star}V_{\lambda^\star}^T + W)^T\left(x - x_{\lambda^\star}^{\mathrm{opt}}\right),$$

where $U_{\lambda^\star}V_{\lambda^\star}^T + W \in \partial\|X\|_*|_{X=\mathcal{A}\left(x_{\lambda^\star}^{\mathrm{opt}}\right)}$, $\langle A, B\rangle = \mathrm{Tr}\, B^T A$ is the standard inner product, and $\mathcal{A}^*$ is the adjoint operator of $\mathcal{A}$. For shorter notation we define

$$\mathcal{A}^*(U_{\lambda^\star}V_{\lambda^\star}^T + W)^T =: a_{\lambda^\star}(W)^T. \quad (5)$$

To sum up, the above inequality becomes

$$\|\mathcal{A}(x)\|_* \geq \left\|\mathcal{A}\left(x_{\lambda^\star}^{\mathrm{opt}}\right)\right\|_* + a_{\lambda^\star}(W)^T\left(x - x_{\lambda^\star}^{\mathrm{opt}}\right), \quad (6)$$

which implies that for $\lambda > \lambda^\star$ the optimal argument x_λ^{opt} must lie in the half-space $\left\{x : a_{\lambda^\star}(W)^T\left(x - x_{\lambda^\star}^{\mathrm{opt}}\right) \leq 0\right\}$.

Using the inequality in (6) we can relax (1) into

$$\begin{aligned}
\min_x \quad & \left\|\mathcal{A}\left(x_{\lambda^\star}^{\mathrm{opt}}\right)\right\|_* + a_{\lambda^\star}(W)^T\left(x - x_{\lambda^\star}^{\mathrm{opt}}\right) \\
\text{s.t.} \quad & \|x - x_o\|_2 \leq \lambda.
\end{aligned} \quad (7)$$

Problem (7) is solved analytically in the following lemma:

Lemma 2.1. *Problem (7) has the optimal solution*

$$x_\lambda^{opt,rlx} = x_o - \frac{\lambda}{\|a_{\lambda^\star}(W)\|_2}a_{\lambda^\star}(W),$$

and optimal cost

$$\left\|\mathcal{A}\left(x_{\lambda^\star}^{opt}\right)\right\|_* + a_{\lambda^\star}(W)^T\left(x_o - x_{\lambda^\star}^{opt}\right) - \lambda\|a_{\lambda^\star}(W)\|_2. \quad (8)$$

Proof. At optimum the constraint is tight and the negative gradient of the cost function, $-a_{\lambda^\star}(W)$, is proportional to the outward pointing normal of the constraint set. This gives $x_\lambda^{opt,rlx}$. Inserting $x_\lambda^{opt,rlx}$ into the cost of (7) gives (8). $\square$

2.2. Bound on cost function approximation error, (3)

Using (8) we can upper bound the approximation error in (3:

Theorem 2.1. *The approximation error in (3) (i.e., the cost function approximation error) for any λ is upper-bounded by the function $d_{\lambda^\star}(\lambda, W)$, as*

$$
\begin{aligned}
&\left\|\mathcal{A}(x_{\lambda^\star}^{opt})\right\|_* - \left\|\mathcal{A}(x_\lambda^{opt})\right\|_* \leq \\
&\lambda \left\|a_{\lambda^\star}(W)\right\|_2 - a_{\lambda^\star}(W)^T \left(x_o - x_{\lambda^\star}^{opt}\right) =: d_{\lambda^\star}(\lambda, W).
\end{aligned} \tag{9}
$$

Proof. The theorem follows from the fact that, for any λ, $\left\|\mathcal{A}(x_\lambda^{\text{opt}})\right\|_*$ is lower bounded by the optimal cost in (8). $\quad\square$

Remark 2.1. *In Section 4 we present a Frank-Wolfe algorithm for optimizing (9) over W. Furthermore, it can be verified that there is some W^{opt} such that $d_{\lambda^\star}(\lambda^\star, W^{opt}) = 0$, by taking $W^{opt} = W^\perp$ according to (13).*

Remark 2.2. *In resemblance with [?], the function $d_{\lambda^\star}(\lambda, W)$ can be interpreted as a duality gap, since the relaxation made in (7) relates to the Frank-Wolfe algorithm [?] when seen as a primal-dual method.*

2.3. Bound on singular value approximation error, (4)

Next, we derive an upper bound on the error in (4). This bound will be the minimum of two separate bounds. The first of these is as follows:

Lemma 2.2.

$$
\sum_i^p \left(\sigma_i^{\lambda^\star} - \sigma_i^\lambda\right)^2 \leq \left\|\sigma^{\lambda^\star} - \left\|\mathcal{A}\left(x_{\lambda^\star}^{opt}\right)\right\|_* e_{i^{min}}\right\|_2^2, \tag{10}
$$

where e_i is the i'th unit vector, i.e., e_i has zeros everywhere except at the i'th component which is one, and $i^{min} = \arg\min \sigma_i^{\lambda^\star}, i = 1, \ldots, p$.

Proof. For $\lambda > \lambda^\star$, $\left\|\mathcal{A}\left(x_\lambda^{\text{opt}}\right)\right\|_* \leq \left\|\mathcal{A}\left(x_{\lambda^\star}^{\text{opt}}\right)\right\|_*$. Hence, (10) corresponds to the maximum of $\sum_i^p \left(\sigma_i^{\lambda^\star} - \sigma_i^\lambda\right)^2$ subject to $\sum_i^p \sigma_i^\lambda \leq \left\|\mathcal{A}\left(x_{\lambda^\star}^{\text{opt}}\right)\right\|_*$, which is reached by making $\sigma_{i^{\min}}$ as large as possible and $\sigma_i = 0$ for $i \neq i^{\min}$. $\quad\square$

Now, we derive a second upper bound, which is complementary to the above. To do this, consider the perturbation

$$
\mathcal{A}\left(x_{\lambda^\star}^{\text{opt}}\right) = \mathcal{A}\left(x_\lambda^{\text{opt}}\right) + E; \quad E := \mathcal{A}\left(x_{\lambda^\star}^{\text{opt}} - x_\lambda^{\text{opt}}\right), \tag{11}
$$

which is valid since $\mathcal{A}$ is linear in x. Then, according to Mirsky's theorem [?] the singular values of $\mathcal{A}(x_\lambda^{\text{opt}})$ obey

$$
\sum_i^p \left(\sigma_i^{\lambda^\star} - \sigma_i^\lambda\right)^2 \leq \|E\|_F^2,
$$

where, due to equivalence of finite-dimensional norms [?],

$$
\|E\|_F^2 = \left\|\mathcal{A}(x_{\lambda^\star}^{\text{opt}} - x_\lambda^{\text{opt}})\right\|_F^2 \leq C_\mathcal{A} \left\|x_{\lambda^\star}^{\text{opt}} - x_\lambda^{\text{opt}}\right\|_2^2, \tag{12}
$$

for some constant $C_\mathcal{A}$ depending on $\mathcal{A}$.

Furthermore, we bound $\left\|x_{\lambda^\star}^{\text{opt}} - x_\lambda^{\text{opt}}\right\|_2^2$ in Lemma 2.4 below. For this we need the following lemma:

Lemma 2.3. *There exists a $W = W^\perp$ such that $a_{\lambda^\star}(W^\perp)$ (see (5)) is proportional to the error vector $\left(x_o - x_{\lambda^\star}^{opt}\right)$, i.e.,*

$$
a_{\lambda^\star}(W^\perp) = \gamma \left(x_o - x_{\lambda^\star}^{opt}\right), \tag{13}
$$

for some scalar $\gamma > 0$.

Proof. The proof is in the Appendix. $\quad\square$

Lemma 2.4.

$$
\left\|x_{\lambda^\star}^{opt} - x_\lambda^{opt}\right\|_2^2 \leq \lambda^2 - (\lambda^\star)^2. \tag{14}
$$

Proof. Due to the existence of $a_{\lambda^\star}(W^\perp)$ in (13), x_λ^{opt} is constrained by the convex set

$$
\mathcal{C}_{\lambda^\star}^\perp(\lambda) := \left\{x : \|x - x_o\|_2 \leq \lambda, a_{\lambda^\star}(W^\perp)^T \left(x - x_{\lambda^\star}^{\text{opt}}\right) \leq 0\right\},
$$

so an upper bound is $\max\limits_{x \in \mathcal{C}_{\lambda^\star}^\perp(\lambda)} \left\|x_{\lambda^\star}^{\text{opt}} - x\right\|_2^2$. This maximum can be solved geometrically. Since $x_{\lambda^\star}^{\text{opt}}$ is inside the ball of the first constraint of $\mathcal{C}_{\lambda^\star}^\perp(\lambda)$, this constraint has to be tight at the optima. Furthermore, with the first constraint being tight, the vectors $(x - x_o)$, $(x_{\lambda^\star}^{\text{opt}} - x_o)$, and $(x_{\lambda^\star}^{\text{opt}} - x)$ form a triangle, with $\|x - x_o\|_2 = \lambda$ and $\left\|x_{\lambda^\star}^{\text{opt}} - x_o\right\|_2 = \lambda^\star$, so

$$
\lambda^2 = \left\|x_{\lambda^\star}^{\text{opt}} - x\right\|_2^2 + (\lambda^\star)^2 - 2\left\|x_{\lambda^\star}^{\text{opt}} - x\right\|_2 \lambda^\star \cos(\frac{\pi}{2} + v)
$$

according to the law of cosines, where $v \geq 0$ is the angle between $x_{\lambda^\star}^{\text{opt}} - x$ and the hyperplane

$$
\left\{x : a_{\lambda^\star}(W^\perp)^T \left(x - x_{\lambda^\star}^{\text{opt}}\right) = 0\right\}.
$$

This expression is maximized for $v = 0$ giving the result $\left\|x_{\lambda^\star}^{\text{opt}} - x^{\text{opt}}\right\|_2^2 = \lambda^2 - (\lambda^\star)^2$. (In fact, $v = 0$ implies that the second constraint in $\mathcal{C}_{\lambda^\star}^\perp(\lambda)$ is also tight.) $\quad\square$

Combining (10), (12), and (14), we obtain the following upper bound on the approximation error in (4):

Theorem 2.2. *The approximation error in (4) is upper bounded by the function $s_{\lambda^\star}(\lambda)$:*

$$
\begin{aligned}
&\sum_i^p \left(\sigma_i^{\lambda^\star} - \sigma_i^\lambda\right)^2 \leq \\
&\leq \min\left\{\left\|\sigma^{\lambda^\star} - \left\|\mathcal{A}\left(x_{\lambda^\star}^{opt}\right)\right\|_* e_{i^{min}}\right\|_2^2, C_\mathcal{A}\left(\lambda^2 - (\lambda^\star)^2\right)\right\} \\
&=: s_{\lambda^\star}(\lambda).
\end{aligned} \tag{15}
$$

Proof. The first argument in the min is given by (10). The second is obtained by combining (12) and (14). $\quad\square$

3. ALGORITHMS

3.1. Model order reduction

In model order reduction, and approximative filter design, the aim is to reduce a high-order model description to a low-order model while preserving the properties according to some fit criterion.

We consider a known Finite Impulse Response (FIR) model of a stable scalar discrete-time linear time-invariant dynamical system, denoted by $g_o \in \mathbb{R}^n$, which is a vector containing its impulse response coefficients. Furthermore, we denote the low-order candidates by g, and consider the H_2 model fit criterion $\|g - g_o\|_2 \leq \lambda$. Note that other criteria commonly used in model order reduction are the H_∞- and Hankel norm-criteria (see [?] or [?]), which are not considered here.

It can be shown [?] that the following Hankel matrix (here taken to be symmetric for simplicity)

$$\mathcal{H}(g) := \begin{bmatrix} g_1 & g_2 & \cdots & g_p \\ g_2 & g_3 & \cdots & g_{p+1} \\ \vdots & \vdots & \ddots & \vdots \\ g_p & g_{p+1} & \cdots & g_n \end{bmatrix}, \quad (16)$$

has the property that its rank is equal to the order (McMillan degree) of the dynamical system which has g as impulse response. This motivates the Hankel matrix rank minimization problem to enforce a low system order.

Using the nuclear norm as surrogate for rank and the H_2 model fit criterion, we formulate the following special case of (1):

$$\begin{aligned} \underset{g}{\text{minimize}} \quad & \|\mathcal{H}(g)\|_* \\ \text{subject to} \quad & \|g - g_o\|_2 \leq \lambda. \end{aligned} \quad (17)$$

Note that in this setting $\sigma(\mathcal{H}(g))$ are the Hankel singular values of the system with g as impulse response.

The adjoint of the Hankel operator in (16), $\mathcal{H}^*(X)$, maps matrices $X \in \mathbb{R}^{p \times p}$ to vectors $x \in \mathbb{R}^n$, by summing the anti-diagonals of X, i.e.,

$$\mathcal{H}^*(X) = x; \quad x_k = \sum_{i+j=k+1} X_{ij}. \quad (18)$$

3.2. The algorithms

The algorithms are outlined in Algorithm 1 and 2. The idea is to adaptively choose a set of discretization points, for which problem (1) is solved. In the intermediate intervals the regularization path is approximated by the previous solution (obtained on the infimum of the current interval). The resulting approximation errors are upper bounded in (9) for Algorithm 1 and (15) for Algorithm 2. The discretization points are chosen as the values of λ for which the upper bound reaches a pre-specified error tolerance, ε. This is visualised in Figure 2.

Note that in Algorithm 1, $d_{\lambda^*}(\lambda, W)$ depends on W. For simplicity, we can set $W = 0$, but in Section 4 we also demonstrate how to optimize $d_{\lambda^*}(\lambda, W)$ over W. Also note that for a Hankel matrix the quantity $\mathcal{C}_\mathcal{A} = n$ satisfies (15).

Algorithms 1 and 2. Approximate regularization paths.
Input: g_o, ε.
Output: Approximate regularization paths such that errors (3) $\leq \varepsilon$ (Alg. 1) or (4) $\leq \varepsilon$ (Alg. 2) for $\lambda = [0, \lambda_{\max}]$.
Initialize $i = 0$. Set $\lambda_0^\star = 0$.
while $\lambda_i^\star \leq \lambda_{\max}$ **do**
  Solve (1) for $\lambda = \lambda_i^\star$, giving $x_{\lambda_i^\star}^{\text{opt}} \rightarrow \sigma^{\lambda_i^\star} = \sigma(\mathcal{A}(x_{\lambda_i^\star}^{\text{opt}}))$.
  Solve $\lambda_{i+1}^\star$ from $d_{\lambda_i^\star}(\lambda_{i+1}^\star, W = 0) = \varepsilon$ (Algorithm 1) or $s_{\lambda_i^\star}(\lambda_{i+1}^\star) = \varepsilon$ (Algorithm 2).
  Accept $x_{\lambda_i^\star}^{\text{opt}}$ as approximate solution for $\lambda = [\lambda_i^\star, \lambda_{i+1}^\star)$.
  Set $i = i + 1$.
end while

3.2.1. Number of evaluations for Algorithm 1

Here we bound the number of evaluations of (1), i.e., the number of iterations of the above algorithm needed to guarantee the error (3) within the tolerance ε.

Theorem 3.1. *The number of evaluations of (1) needed by Algorithm 1 is at most*

$$M_{Alg1} \leq \left\lfloor \frac{2 c_n \|g_o\|_2}{\varepsilon} \right\rfloor = \mathcal{O}(\varepsilon^{-1}), \quad (19)$$

in general, and if $W = 0$:

$$M_{Alg1}^{W=0} \leq \left\lfloor \frac{c_n \|g_o\|_2}{\varepsilon} \right\rfloor = \mathcal{O}(\varepsilon^{-1}), \quad (20)$$

where ε is the tolerance and

$$c_n := \left(2 \sum_{k=1}^{p-1} k^2 + p^2 \right)^{\frac{1}{2}} = \|\mathcal{H}^*(\mathbb{1}_{p \times p})\|_2,$$

in which $n = 2p - 1$, $\mathbb{1}_{p \times p}$ is a $(p \times p)$-matrix of ones, and the adjoint of the Hankel operator is defined in (18).

Proof. The proof is in the Appendix. $\square$

3.2.2. Number of evaluations for Algorithm 2

Now, we bound the number of evaluations of (1), i.e., the number of iterations of Algorithm 2 needed to guarantee the solution within the tolerance ε.

Theorem 3.2. *The number of evaluations of (1) needed by Algorithm 2, i.e., the number of iterations, is at most*

$$M_{Alg2} \leq \left\lfloor \frac{n \|g_o\|_2^2}{\varepsilon} \right\rfloor = \mathcal{O}(\varepsilon^{-1}). \quad (21)$$

Proof. The proof is in the Appendix. $\square$

benchmark	order	T_s	n	cpu ADMM	$\varepsilon^{\min}/J^{\max}$	Algorithm 1			Algorithm 2	
						$\varepsilon/J^{\max}$	m	$M^{W=0}$	M	m
beam.mat	348	1	1047	134.23	0.1233	0.2	5	3062	30	10
						0.3	3	4082	20	5
build.mat	48	0.025	576	317.86	0.1607	0.2	7	1035	30	10
						0.3	4	1379	20	5
eady.mat	598	0.1	196	7.36	0.0958	0.2	5	643	30	9
						0.3	3	856	20	4
heat $-$ cont.mat	200	0.5	139	11.78	0.7270	0.2	5	482	30	12
						0.3	3	642	20	7
pde.mat	84	0.0001	242	39.80	0.1054	0.2	5	1193	30	7
						0.3	3	795	20	3

Table 1. Results of Algorithm 1 and 2. T_s is sampling time in Matlab's `c2d`, giving impulse response lengths n. 'cpu ADMM' is an average time in seconds with a standard laptop for solving (17) using ADMM. The maximum cost $J^{\max} := \|\mathcal{H}(g_o)\|_*$. m is number of grid points needed, with upper bounds M (for Algorithm 1 we use (20)). $\varepsilon^{\min}$ is the minimum tolerance for which $d_{\lambda^\star}(\lambda^\star, W = 0) < \varepsilon^{\min}$ for all $\lambda^\star$. For Algorithm 2, $\varepsilon = n \|g_o\|_2^2 / M_{\mathrm{Alg2}}$.

4. IMPLEMENTATION

For large scale problems (1) we suggest an Alternating Direction Method of Multipliers (ADMM), c.f. [?] and [?]. We will follow the method in [?] with a modification for the g-update in (24) below.

First, we rewrite (17) as

$$\begin{aligned} \underset{g\in\mathbb{R}^n, H\in\mathbb{R}^{p\times p}}{\text{minimize}} \quad & \|H\|_* \\ \text{subject to} \quad & \|g - g_o\|_2 \leq \lambda \\ & \mathcal{H}(g) = H. \end{aligned} \quad (22)$$

Next, we form the following augmented Lagrangian

$$L_\rho(H, g, Z) =$$
$$\|H\|_* + \mathrm{Tr}\left(Z^T(\mathcal{H}(g) - H)\right) + \frac{\rho}{2}\|\mathcal{H}(g) - H\|_F^2.$$

The strategy is to update the variables as

$$H^{k+1} := \arg\min_H L_\rho(H, g^k, Z^k) \quad (23)$$

$$g^{k+1} := \underset{\{g : \|g-g_o\|_2 \leq \lambda\}}{\arg\min} L_\rho(H^{k+1}, g, Z^k) \quad (24)$$

$$Z^{k+1} := Z^k + \rho(\mathcal{H}(g^{k+1}) - H^{k+1}). \quad (25)$$

The variables can be initialized e.g. as $H = 0, g = 0, Z = 0, \rho = 1$. (Initialize ρ if it is adaptive as in [?]).

The H update in (23) is accomplished in [?] using so called 'singular value soft-thresholding':

$$H^{k+1} = \arg\min_H L_\rho(H, g^k, Z^k)$$
$$= \arg\min_H \left(\|H\|_* + \frac{\rho}{2}\left\|H - \mathcal{H}(g^k) - (1/\rho)Z^k\right\|_F^2 \right)$$
$$= \sum_{i=1}^p \max\left\{0, \sigma_i - \frac{1}{\rho}\right\} u_i v_i^T,$$

where σ_i, u_i, v_i are given by the singular value decomposition

$$\mathcal{H}(g^k) + \frac{1}{\rho}Z^k = \sum_{i=1}^p \sigma_i u_i v_i^T.$$

The second subproblem, (24), we reformulate as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \frac{\rho}{2}x^T P x + q^T x \\ \text{subject to} \quad & \|x\|_2 \leq \lambda, \end{aligned}$$

where $x = g - g_o$, $P = \mathrm{diag}(\mathcal{H}^*(\mathbb{1}_{p\times p}))$ and $q = \mathcal{H}^*(Z^k + \rho\mathcal{H}(g_o) - \rho H^{k+1})$, and $\mathcal{H}^*(\cdot)$ is defined in (18). This can be solved by using the facts that the optimal point, x^{opt}, lies on the boundary of the constraint set, and in this point the negative gradient of the cost function is normal to the constraint set, i.e., it is proportional to x^{opt}. This means that

$$\rho P x^{\mathrm{opt}} + q = -t x^{\mathrm{opt}} \Leftrightarrow x^{\mathrm{opt}} = -(\rho P + t I)^{-1} q$$

where $t \geq 0$ is a scalar determined from solving $f(t) := \|x^{\mathrm{opt}}\|_2 = \lambda$ using Newton's method. This t is unique since

$$f(t) = \left\|(\rho P + t I)^{-1} q\right\|_2 = \left(\sum_{i=1}^n \frac{q_i^2}{(t + \rho P_{ii})^2}\right)^{\frac{1}{2}},$$

is a decreasing function with $f(0) > \lambda$ and $f(\infty) = 0$. The fact that $f(0) > \lambda$ is true since $x^{\mathrm{opt}}(t = 0)$ is the global minimum, which is located outside the constraint set. Summing up, we obtain

$$g^{k+1} = g_o - (\rho P + t I)^{-1} q. \quad (26)$$

The stopping criterion is $\|r_{\mathrm{p}}^{k+1}\| \leq \epsilon_{\mathrm{p}}^{k+1}$ and $\|r_{\mathrm{d}}^{k+1}\| \leq \epsilon_{\mathrm{d}}^{k+1}$, where the primal and dual residuals (r_{p} and r_{d}) and tolerances (ϵ_{p} and ϵ_{d}) are computed from the definition in [?,

Sec. 3] as

$$r_{\text{p}}^{k+1} := \mathcal{H}(g^{k+1}) - H^{k+1}$$
$$r_{\text{d}}^{k+1} := \rho \mathcal{H}^*(H^k - H^{k+1})$$
$$\epsilon_{\text{p}}^{k+1} := p\epsilon_{\text{abs}} + \epsilon_{\text{rel}} \max \left\{ \left\| \mathcal{H}(g^{k+1}) \right\|_F, \left\| H^{k+1} \right\|_F \right\}$$
$$\epsilon_{\text{d}}^{k+1} := \sqrt{n}\epsilon_{\text{abs}} + \epsilon_{\text{rel}} \left\| \mathcal{H}^*(Z^{k+1}) \right\|_2 .$$

4.1. Frank-Wolfe algorithm for optimizing (9) over W

The Frank-Wolfe algorithm (or *conditional gradient method*) is a simple iterative method, suggested in [**?**] (1956) for minimizing convex, continuously differentiable functions f over compact convex sets. We here design a Frank-Wolfe algorithm for optimizing (9) over W. Our algorithm is summarized in Algorithm 3.

To solve the argument minimizations at each iteration explicitly, we note that for the constraints in $\mathcal{M}$

$$U^T W = 0 \Leftrightarrow W = U^\perp A,$$

and

$$WV = U^\perp AV = 0 \Leftrightarrow AV = 0 \Leftrightarrow A = D(V^\perp)^T,$$

for some matrices A and D of appropriate size. Hence,

$$W = U^\perp D(V^\perp)^T,$$

where $\|W\| \leq 1 \Rightarrow \|D\| \leq 1$. Then, in Algorithm 3, we parameterize $X = U^\perp D(V^\perp)^T$, so that $\langle X, C \rangle = \langle U^\perp D(V^\perp)^T, C \rangle = \langle D, (U^\perp)^T C V^\perp \rangle =: \langle D, \tilde{C} \rangle$, and solve

$$\arg \min_{\|D\| \leq 1} \langle D, \tilde{C} \rangle,$$

This problem has the closed form solution $D^{\text{opt}} = U_{\tilde{C}} V_{\tilde{C}}^T$ where $\tilde{C} = U_{\tilde{C}} \Sigma_{\tilde{D}} V_{\tilde{C}}^T$ is a compact singular value decomposition. Then,

$$X^{\text{opt}} = U^\perp D^{\text{opt}}(V^\perp)^T = U^\perp U_{\tilde{C}} V_{\tilde{C}}^T (V^\perp)^T.$$

Finally, when optimizing the duality gap (9) over W for a fixed λ, we have

$$C = \left. \frac{\partial d_{\lambda^\star}(\lambda, W)}{\partial W} \right|_{W=W^k} = \frac{\lambda}{\|h_{\lambda^\star}\|_2} \mathcal{H}(h_{\lambda^\star}) + \mathcal{H}\left(x_{\lambda^\star}^{\text{opt}} - g_o \right).$$

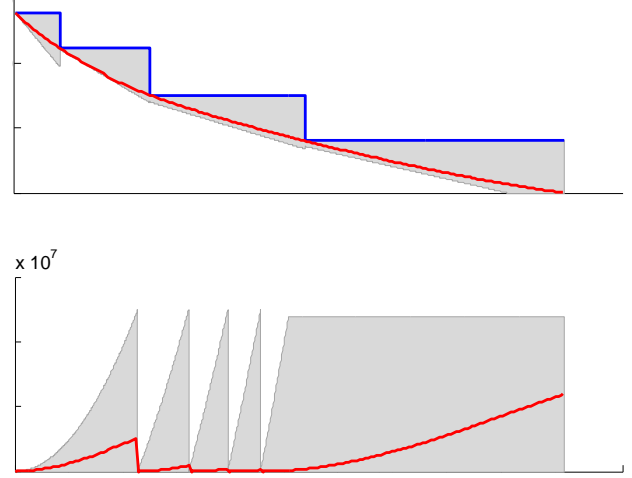**Algorithm 3.** Frank-Wolfe for optimizing (9) over W
Initialize $W^0 = 0$
for k = 0,1,...,K **do**
     Compute $X^{\text{opt}} := \langle X, C \rangle;\ C = \left. \frac{\partial d_{\lambda^\star}(\lambda, W)}{\partial W} \right|_{W=W^k}$
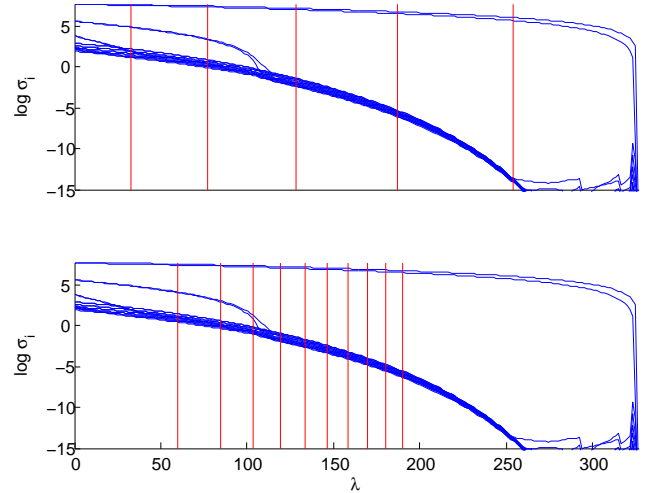 $X \in \mathcal{M}$
     Update $W^{k+1} := (1 - \gamma)W^k + \gamma X^{\text{opt}}$, for $\gamma = \frac{2}{2+k}$
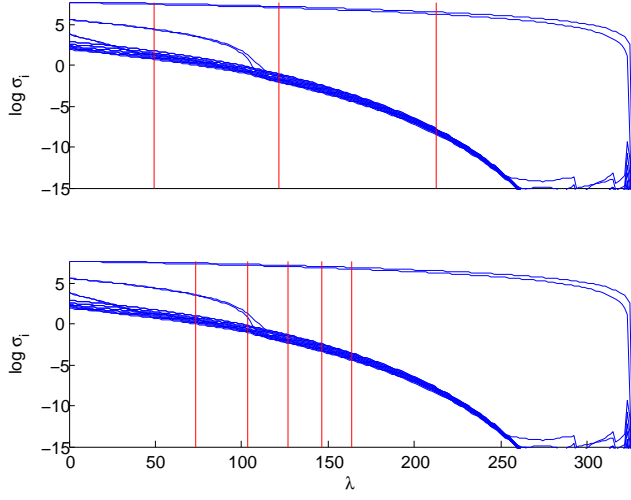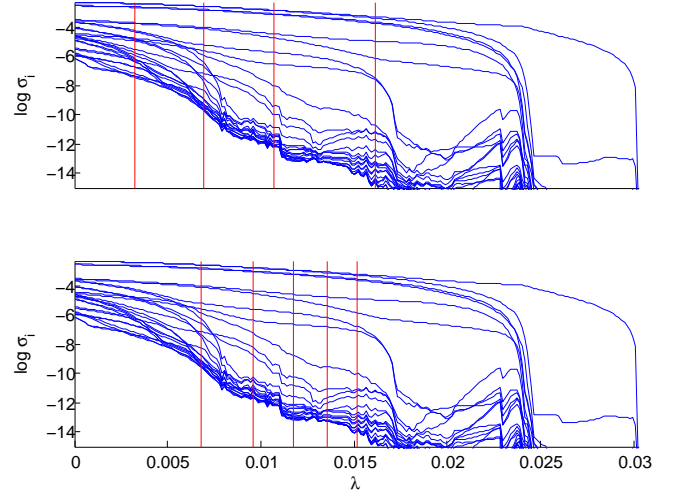**end for**

Fig. 2. True errors (red) and confidence zones (grey) for the model `build.mat`. Upper: Algorithm 1 with approximate regularization path (blue) for $\varepsilon/J^{\max} = 0.3$, where $J^{\max} = \|\mathcal{H}(g_o)\|_*$. Lower: Algorithm 2 for $\varepsilon = n \|g_o\|_2^2 /M$, where $M = 30$.
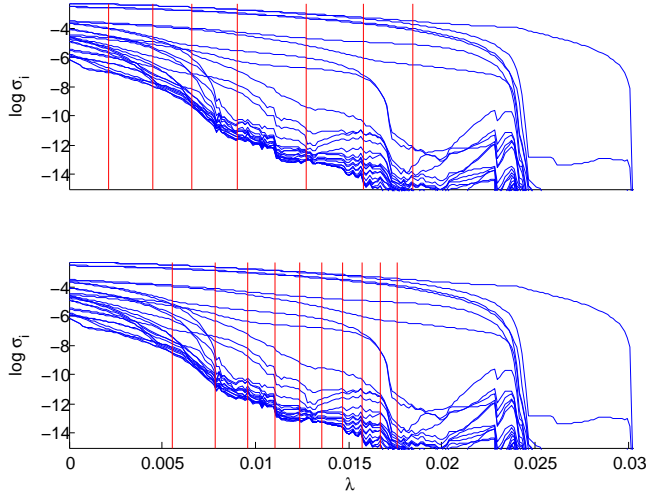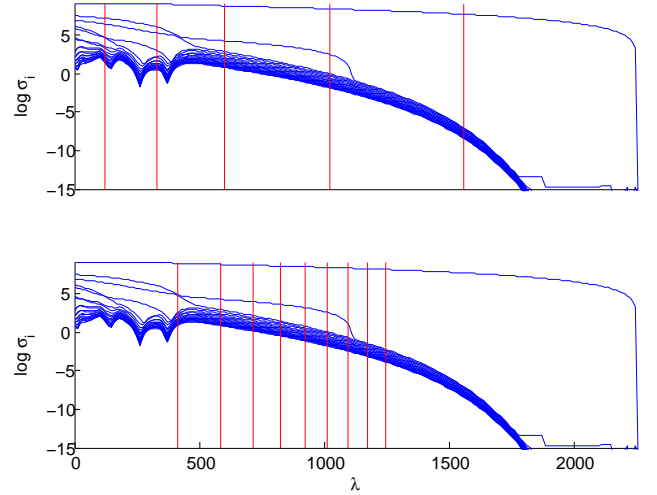

Fig. 3. Plot of significant singular values $\sigma_i, i = 1, \ldots, 17$, for `beam.mat`. Vertical lines indicate grid points. Upper: Algortihm 1 ($\varepsilon/J^{\max} = 0.2$, where $J^{\max} = \|\mathcal{H}(g_o)\|_*$). Lower: Algorithm 2 ($\varepsilon = n \|g_o\|_2^2 /M$, where $M = 40$).

Fig. 4. Plot of significant singular values $\sigma_i, i = 1, \ldots, 17$, for `beam.mat`. Vertical lines indicate grid points. Upper: Algortihm 1 ($\varepsilon/J^{\max} = 0.3$, where $J^{\max} = \|\mathcal{H}(g_o)\|_*$). Lower: Algorithm 2 ($\varepsilon = n \|g_o\|_2^2 /M$, where $M = 30$).
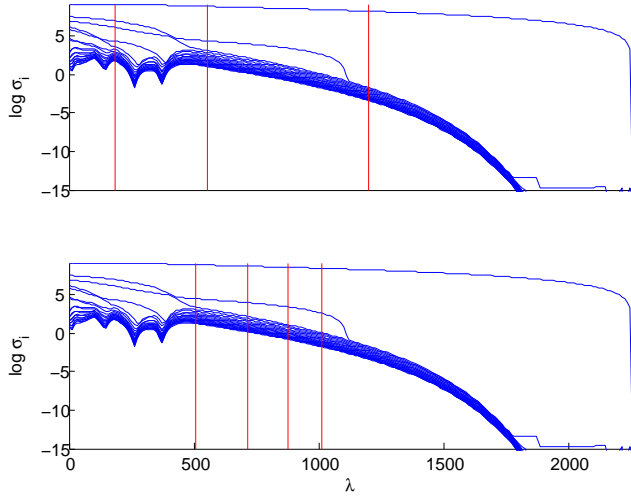

Fig. 6. Plot of significant singular values $\sigma_i, i = 1, \ldots, 26$, for `build.mat`. Vertical lines indicate grid points. Upper: Algortihm 1 ($\varepsilon/J^{\max} = 0.3$, where $J^{\max} = \|\mathcal{H}(g_o)\|_*$). Lower: Algorithm 2 ($\varepsilon = n \|g_o\|_2^2 /M$, where $M = 30$).
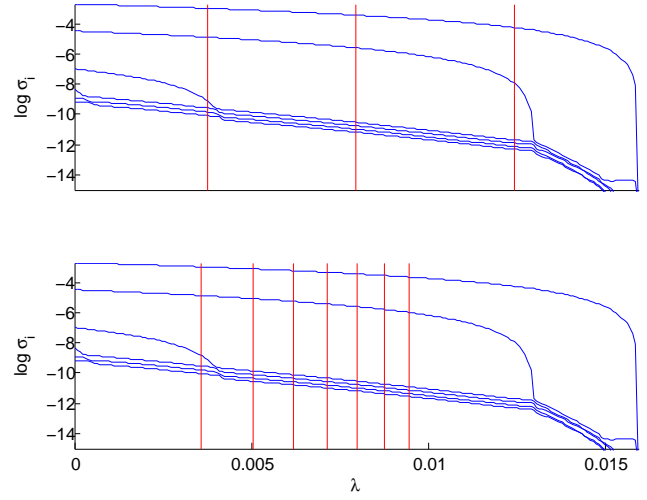

Fig. 5. Plot of significant singular values $\sigma_i, i = 1, \ldots, 26$, for `build.mat`. Vertical lines indicate grid points. Upper: Algortihm 1 ($\varepsilon/J^{\max} = 0.2$, where $J^{\max} = \|\mathcal{H}(g_o)\|_*$). Lower: Algorithm 2 ($\varepsilon = n \|g_o\|_2^2 /M$, where $M = 40$).

Fig. 7. Plot of significant singular values $\sigma_i, i = 1, \ldots, 20$, for `eady.mat`. Vertical lines indicate grid points. Upper: Algortihm 1 ($\varepsilon/J^{\max} = 0.2$, where $J^{\max} = \|\mathcal{H}(g_o)\|_*$). Lower: Algorithm 2 ($\varepsilon = n \|g_o\|_2^2 /M$, where $M = 40$).
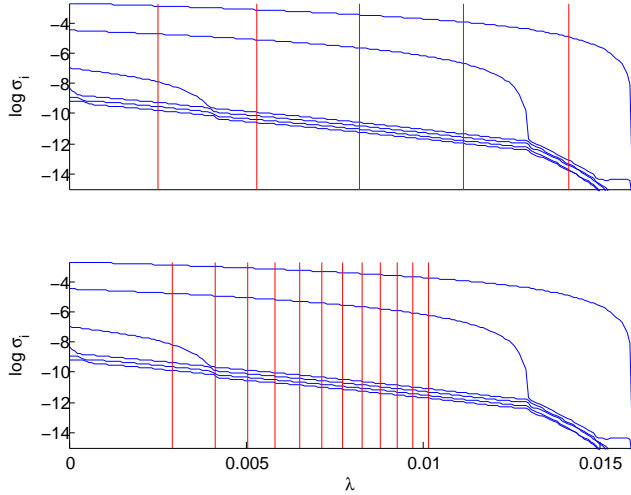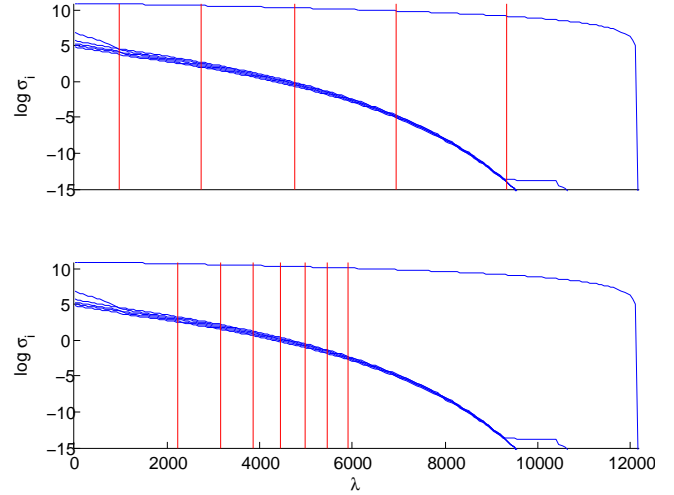
Fig. 8. Plot of significant singular values $\sigma_i, i = 1, \ldots, 20$, for `eady.mat`. Vertical lines indicate grid points. Upper: Algortihm 1 ($\varepsilon/J^{\max} = 0.3$, where $J^{\max} = \|\mathcal{H}(g_o)\|_*$). Lower: Algorithm 2 ($\varepsilon = n\|g_o\|_2^2/M$, where $M = 30$).

Fig. 10. Plot of significant singular values $\sigma_i, i = 1, \ldots, 6$, for `heat − cont.mat`. Vertical lines indicate grid points. Upper: Algorithm 1 ($\varepsilon/J^{\max} = 0.3$, where $J^{\max} = \|\mathcal{H}(g_o)\|_*$). Lower: Algorithm 2 ($\varepsilon = n\|g_o\|_2^2/M$, where $M = 30$).
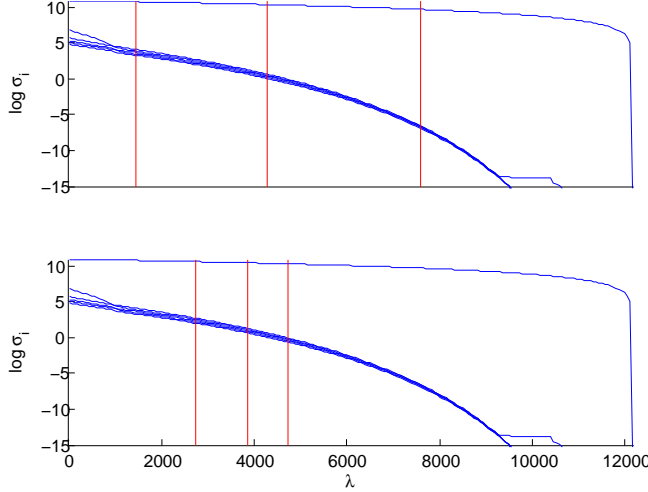

Fig. 9. Plot of significant singular values $\sigma_i, i = 1, \ldots, 6$, for `heat − cont.mat`. Vertical lines indicate grid points. Upper: Algortihm 1 ($\varepsilon/J^{\max} = 0.2$, where $J^{\max} = \|\mathcal{H}(g_o)\|_*$). Lower: Algorithm 2 ($\varepsilon = n\|g_o\|_2^2/M$, where $M = 40$).

Fig. 11. Plot of significant singular values $\sigma_i, i = 1, \ldots, 6$, for `pde.mat`. Vertical lines indicate grid points. Upper: Algortihm 1 ($\varepsilon/J^{\max} = 0.2$, where $J^{\max} = \|\mathcal{H}(g_o)\|_*$). Lower: Algorithm 2 ($\varepsilon = n\|g_o\|_2^2/M$, where $M = 40$).

Fig. 12. Plot of significant singular values $\sigma_i, i = 1, \ldots, 6$, for `pde.mat`. Vertical lines indicate grid points. Upper: Algortihm 1 ($\varepsilon/J^{\max} = 0.3$, where $J^{\max} = \|\mathcal{H}(g_o)\|_*$). Lower: Algorithm 2 ($\varepsilon = n\|g_o\|_2^2/M$, where $M = 30$).

5. RESULTS

Algorithm 1 and 2 are implemented on single-input-single-output model order reduction benchmarks[1]. The continuous-time impulse response sequences are discretized using the Matlab command `c2d` with sampling times listed in Table 1. We here set $W = 0$, but design a Frank-Wolfe algorithm for optimizing (9) over W in Section 4. The tolerances ε are chosen as a fraction the maximum possible cost of (17) in Algorithm 1, and according to $\varepsilon = n\|g_o\|_2^2/M_{\mathrm{Alg2}}$ for Algorithm 2.

For large scale problems (1) we suggest an Alternating Direction Method of Multipliers (ADMM), c.f. [?], [?]. Our method is similar to [?] and provided in Section 4.

The results are summarized in Table 1. We observe that, for Algorithm 1, the bound M_{Alg1} is very loose. We also see the smallest possible ε such that $d_{\lambda^\star}(\lambda^\star, W = 0) < \varepsilon$ for any $\lambda^\star \in (\lambda_{\min}, \lambda_{\max})$. For the system `heat − cont.mat` this extreme value is high, but we observe that for most part of the regularization path $d_{\lambda^\star}(\lambda^\star, W = 0)$ is very low; it only increases very close to $\lambda_{\max}$. For smaller values of ε in Algorithm 1, we may optimize over W. Then, it is possible to use arbitrarily small ε, since $d_{\lambda^\star}(\lambda^\star, W^\perp) = 0$ (see Remark 2.1).

In Figure 2 we illustrate the ideas in Algorithm 1 and 2.

In Figures 3-12 we visualize the grid points for Algorithm 1 and 2, respectively, when applied to the model the different benchmark models. For Algorithm 1, we generally see that the grid points are slightly more dense in for smaller values of λ. For Algorithm 2, the grid points are also more dense in the

---

[1]The model reduction benchmarks are avaliable at slicot.org/20-site/126-benchmark-examples-for-model-reduction

first part of the regularization path; it stops gridding when the first argument in (15) becomes active. Thus, the algorithms are suitable in cases where the singular value drops happen for low values of λ, which is a general observation we have made when studying these benchmark examples.

6. CONCLUSION

We have proposed a method to approximate the regularization path of a quadratically constrained nuclear norm minimization problem, with guarantees on the singular values of the matrix argument. The algorithms solve the problem for a finite, explicitly upper-bounded, number of values of the regularization parameter. We have also provided details regarding efficient implementation of the algorithms.

The results show that the algorithms generate grid points that are suitable for tracking changes in the singular values.

7. APPENDIX

Proof of Lemma 2.3. The existence of vectors proportional to $x_o - x_{\lambda^\star}^{\mathrm{opt}}$ can be proved using the optimality conditions of (1), which imply that the minimizer of (1), x_λ^{opt}, minimizes the Lagrangian $L(g, z)$ for some Lagrange multiplier $z \geq 0$ [?, pp. 217], i.e.,

$$
\begin{aligned}
0 &\in \partial_x \left. L(x, z) \right|_{x = x_{\lambda^\star}^{\mathrm{opt}}} \\
&= \partial_x \left. \|\mathcal{A}(x)\|_* \right|_{x = x_{\lambda^\star}^{\mathrm{opt}}} + \partial_x z \left. \|x - x_o\|_2 \right|_{x = x_{\lambda^\star}^{\mathrm{opt}}}.
\end{aligned}
$$

The second term is computed explicitly as

$$
\partial_x z \left. \|x - x_o\|_2 \right|_{x = x_{\lambda^\star}^{\mathrm{opt}}} = \frac{z}{\left\| x_{\lambda^\star}^{\mathrm{opt}} - x_o \right\|_2} \left(x_{\lambda^\star}^{\mathrm{opt}} - x_o \right),
$$

so that

$$
\frac{z}{\left\| x_{\lambda^\star}^{\mathrm{opt}} - x_o \right\|_2} \left(x_o - x_{\lambda^\star}^{\mathrm{opt}} \right) \in \partial_x \left. \|\mathcal{A}(x)\|_* \right|_{x = x_{\lambda^\star}^{\mathrm{opt}}},
$$

showing that there is a subgradient proportional to $x_o - x_{\lambda^\star}^{\mathrm{opt}}$. $\square$

Proof of Theorem 3.1. Consider the choice of $h_{\lambda^\star}(W) = h_{\lambda^\star}(W^\perp)$ in (13). Then, for $\lambda > \lambda^\star$,

$$
d_{\lambda^\star}\left(\lambda, W^{\mathrm{opt}}\right) \leq d_{\lambda^\star}\left(\lambda, W^\perp\right) = \left\| h_{\lambda^\star}\left(W^\perp\right) \right\|_2 \left(\lambda - \lambda^\star\right),
$$

in which we will now bound $\left\| h_{\lambda^\star}\left(W^\perp\right) \right\|_2$. To do this, we

can bound

$$\|h_{\lambda^{\star}}(W)\|_2 = \left(\sum_{k=1}^{n}\left(\mathrm{Tr}\,(UV^T+W)^T H_k\right)^2\right)^{\frac{1}{2}} =$$

$$\left(\sum_{k=1}^{n}\left(\mathrm{Tr}\,VU^T H_k + \mathrm{Tr}\,W^T H_k\right)^2\right)^{\frac{1}{2}} \leq \left(\sum_{k=1}^{n}\left(2\,\|H_k\|_*\right)^2\right)^{\frac{1}{2}}$$

$$= 2\left(\sum_{k=1}^{p}\|I_k\|_*^2 + \sum_{k=p+1}^{n}\|I_{n-k+1}\|_*^2\right)^{\frac{1}{2}} = 2c_n,$$

(27)

where we have used in the first inequality twice the characterization of the nuclear norm as

$$\|X\|_* = \sup\left\{\mathrm{tr}(Y^T X) : \|Y\| \leq 1\right\}$$

with $Y = UV^T$ such that $\left\|UV^T\right\| = 1$, and $Y = W$ such that $\|W\| \leq 1$, and in the second last equality the unitary invariance of the nuclear norm.

Using the bound on $\|h_{\lambda^{\star}}(W)\|_2$ in $\left\|h_{\lambda^{\star}}\left(W^{\perp}\right)\right\|_2 (\lambda - \lambda^{\star})$ the sub-intervals of the algorithm are of length at most

$$\lambda_{i+1}^{\star} - \lambda_i^{\star} = \frac{\varepsilon}{2c_n}.$$

Hence, we need at least

$$\left\lfloor \frac{2c_n(\lambda_{\max} - \lambda_{\min})}{\varepsilon} \right\rfloor$$

evaluations of (1). Since $(\lambda_{\min}, \lambda_{\max}) = (0, \|g_o\|_2)$ according to (2), we obtain (19). If we set $W = 0$ in (27) we obtain (20). $\qquad\square$

Proof of Theorem 3.2. If the first argument in (15) is ignored Algorithm 2 will use a greater or equal number of grid points. In this case, Algorithm 2 evaluates (1) for $\lambda = \lambda_i^{\star}, i = 1, \ldots, M_{\mathrm{Alg2}}$, obtained from $s_{\lambda_i^{\star}}(\lambda_{i+1}^{\star}) = \varepsilon$. As in (15), $s_{\lambda^{\star}}(\lambda) = n\left(\lambda^2 - (\lambda^{\star})^2\right)$, which gives the relation

$$\lambda_{i+1}^{\star} = \left(\frac{1}{n}\varepsilon + (\lambda_i^{\star})^2\right)^{1/2}.$$

By induction in i, with $\lambda_0^{\star} = 0$, this becomes

$$\lambda_{i+1}^{\star} = \left(\frac{i}{n}\varepsilon\right)^{1/2}.$$

Given that $\lambda_{\max} = \|g_o\|_2$, as in (2), the largest integer M_{Alg2} such that $\left(\frac{M_{\mathrm{Alg2}}}{n}\varepsilon\right)^{1/2} \leq \|g_o\|_2$ obeys $M_{\mathrm{Alg2}} = \left\lfloor \frac{n\|g_o\|_2^2}{\varepsilon} \right\rfloor$. Since this is a worst-case scenario we obtain the upper bound in (21). $\qquad\square$